David Kirby

ECE 529: Introduction to Technical Cybersecurity

Spring, 2022

# Binary Analysis

Binary analysis can tell us an incredible amount of details about a program. Even though this executable was compiled for x86, using Hopper, I was able to open the `msg` executable directly, without having to disassemble the code first. From there I was able to see that the first string was hard-coded in the `main()` program – string 1: **`fubar`** (see Figure 1).

```
        ; =============== B E G I N N I N G   O F   P R O C E D U R E ===============

        ; Variables:
        ;    var_4: int32_t, -4
        ;    var_10: int64_t, -16


                    main:
0000000000000972        push    rbp                                     ; End of unwind block (FDE at 0xb1c), Begin
0000000000000973        mov     rbp, rsp
0000000000000976        sub     rsp, 0x10
000000000000097a        mov     dword [rbp+var_4], edi
000000000000097d        mov     qword [rbp+var_10], rsi
0000000000000981        lea     rax, qword [aFubar]                     ; "fubar"
0000000000000988        mov     qword [MSG], rax                        ; MSG
000000000000098f        mov     eax, 0x0
0000000000000994        call    print_msg                               ; print_msg
0000000000000999        mov     eax, 0x0
000000000000099e        mov     edi, eax                                ; argument "__status" for method j_exit
00000000000009a0        call    j_exit                                  ; exit
                      ; endp
00000000000009a5        align   16                                      ; End of unwind block (FDE at 0xb40)
```

**FIGURE 1: STRING 1 – FUBAR.**

To run the executable, I needed to use the x86 Docker container I created for Module 3 of this course. The second string was generated when the program was executed with the shell command uname – string 2: **`Linux`** (see Figure 2). The third string was discovered by setting break points after the operations were performed – string 3: **`J<,4*`**. Finally, the fourth string was revealed after running the program – string 4: **SE5=3**.

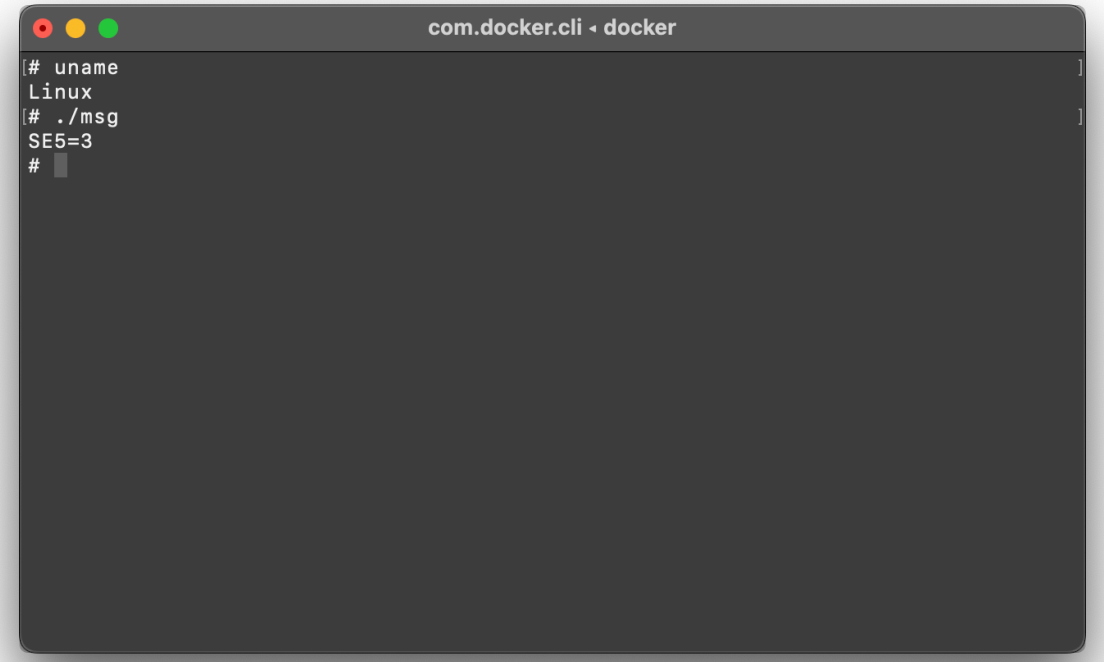


**FIGURE 2: STRINGS 2 & 4 – LINUX & SE5=3.**